

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

| | | | |
|------------|---|-------------------|---------------------|
| Applicant | : Taher ELGAMAL et al. | Art Unit | : 2135 |
| Serial No. | : 09/920,801 | Examiner | : Klimach, Paula W. |
| Filed | : August 3, 2001 | Confirmation No.: | 8214 |
| Title | : CRYPTOGRAPHIC POLICY FILTERS AND POLICY CONTROL METHOD AND APPARATUS | | |

REPLY TO FINAL OFFICE ACTION OF FEBRUARY 8, 2006

Remarks/Arguments begin on page 2 of this paper.

REMARKS

In response to the final office action of February 8, 2006, applicants asks that all claims be allowed in view of the following remarks.

Claims 31-34, 36-43 and 45-48 are now pending, of which claims 31 and 40 are independent. No claims have been amended or added. No new matter has been added.

Drews in View of Perona Rejection

The Final Office Action rejected claims 31, 36-40, and 45-45 under 36 U.S.C. 103 as being unpatentable over Drews (U.S. Patent No. 6,647,494) in view of Perona (U.S. Patent No. 6,671,809). Applicants submit that independent claims 31 and 40 each define an invention that is patentable over Drews, Perona, or any valid combination of the references, as described more fully below.

Independent Claim 31 dependent claims 36-39

Claim 31 is directed to a method for controlling functions of an application program. The method includes accessing a policy file that includes an attribute portion configured to store one or more policy attributes and a value portion having one or more attribute values. Each attribute value corresponds to a policy attribute and indicates whether an application program may access the function represented by the policy attribute. Each policy file includes a signature portion with at least one digital certificate. The method includes determining whether the policy file is unaltered based on the signature portion of the policy file. The method also includes retrieving at least one of the attributes, and, for each retrieved attribute, an attribute value corresponding to the attribute from the policy file. The method further includes determining whether a function represented by a retrieved attribute is permitted to be accessed by the application program and permitting the application program to access the function conditioned upon a determination that the policy file is unaltered.

Applicants submit that neither Drews, Perona, nor the proposed combination of Drews and Perona discloses or suggests the features of claim 31. In particular, neither Drews, Perona, nor any valid combination of the references describe or suggest accessing a policy file that includes an attribute portion configured to store one or more policy attributes, each of which represents a function capable of being performed by the application program, nor do the

references describe or suggest determining whether a function represented by a retrieved attribute is permitted to be accessed by the application program.

Beginning with a discussion of Drews, the Final Office Action states:

Drews does not disclose expressly disclose [sic] determining whether an application program may use a function capable of being performed by the application program and thus determining whether a function represented by a received attribute is permitted to be accessed by the application program; and permitting the application program to access the function conditioned upon the determination that the policy file is unaltered. [See page 3]

Applicants acknowledge these shortcomings of Drews with particular reference to the claim 31 requirement for “each attribute value corresponding to a policy attribute and indicating whether an application program may use a function capable of being performed by the application program” and “determining whether a function represented by a retrieved attribute is permitted to be accessed by the application program.” Applicants respectfully note that the language of the Final Office Action recited above does not match the claim language with respect to claim 31.

Turning to the secondary reference, the Final Office Action states

Perona discloses determining whether an application program may use a function capable of being performed by the application program and thus determining whether a function represented by a retrieved attribute is permitted to be accessed by the application program (column 6 lines 15-23); and permitting the application program to access the function conditioned upon the determination that the policy file is unaltered (column 6 lines 24-36). [See page 3]

Applicants respectfully disagree. Applicants submit Perona does not disclose or suggest the claim language recited in claim 31. Moreover, it should be noted that language of the Final Office Action (cited above) does not match the claim language with respect to claim 31.

Perona generally relates to open architecture software that enables software to be loaded on a platform, such as a cellular phone, by performing checks among system components based on configuration and rule information.¹ In Perona, software modules may be installed to update or add to an application on a platform. The software modules include requirements that “must be met by the modules” for the platform to be able to load the modules.² In particular, the software modules include pointer record rules specifying requirements and limitations imposed on the

¹ See Perona, Column 1, lines 7-13 and FIG. 1

² See Perona, Column 4, lines 14-16

referenced module, such as, for example, interoperability information.³ If the rules are met, the platform is able to load the module for execution.⁴ In one example, an RF module requires the platform include a specific hardware component to enable loading.⁵

As such, Perona simply discloses a method of analyzing requirements in a software module in order to establish whether the platform is capable of loading a module.⁶ In Perona, examples are given where a platform uses rules included in a module to determine if requirements are met.⁷ By contrast, Applicant's claim includes "determining whether a function represented by a retrieved attribute is permitted to be accessed by the application program."

Additionally, Perona does not disclose or suggest "each attribute value corresponding to a policy attribute and indicating whether an application program may use a function capable of being performed by the application program," as recited in claim 31. Perona simply discloses requirements that dictate whether a platform is able to load a module. Alternatively, Applicant's claim 31 includes attribute values that indicate whether an application is permitted to use a function that the application is capable of using, without regard for compatibility issues addressed by Perona. Thus, as illustrated by dependent claim 38, claim 31's method may be used to determine permission by including a truth expression in each of the attribute values wherein the truth expression is one of a true flag, a false flag, and a conditional flag which may enable, disable, or conditionally enable a function of a program.

Perona deals with a process for confirming compatibility among software modules to avoid the downloading and execution of incompatible modules. Perona maintains interoperability rules that are consulted as a condition precedent to loading and executing software modules. For example, Perona's interoperability rules check to see whether necessary software is loaded on a target machine based on satisfaction of the rule.

Perona's rules therefore relate to the existence or absence of compatible or necessary software on a target computer. While Perona's rules dictate whether or not a target computer can load or use an application, they do not relate to or otherwise "indicating whether an application program may use a function capable of being performed by the application program" or

³ See Perona, Column 4, lines 26-32

⁴ See Perona, Column 4, lines 37-43

⁵ See Perona, Column 5, lines 18-22

⁶ See Perona, Column 4, lines 37-43

⁷ See Perona, Column 5, lines 18-22

determine “whether a function represented by a retrieved attribute is permitted to be accessed by the application program” as recited by claim 31 (emphasis added).

More specifically, Perona teaches rules that enable a binary decision – whether or not to load software onto a target computer. Clearly, this decision informs whether or not the target computer has access to the software to be loaded thereupon, as satisfaction of Perona's rules and resultant loading of a software application avails the target computer to the software application and all of its functionality, and dissatisfaction of Perona's rules inhibits loading of the software application and thus denies the target computer access to all of the software application's functionality. It is equally clear that the binary decision enabled by Perona's rules pertains to all of the functions of each particular software application, enabling or denying access to all functions of a software application collectively, without discrimination. As such, Perona's rules regulate access by a target computer to a software application and its functions, either granting the target computer unfettered access to all of the software application's function by allowing the software application to be downloaded, or by denying the target computer access to any of the software application's functions by preventing its downloading.

Moreover, Perona does not discriminate among any of a software application's functions. Thus, Perona, does not regulate or even indicate whether the application program itself may use one of its functions. This distinction, between Perona's compatibility rules that control a target computer's access to an application (and thus all of its functions, indiscriminately) and the claimed requirement for attribute values that indicate (and thus control) an application's program's use of one of its own functions, may be readily understood with reference to the following example. Assume the existence of an application program that is used to control encryption, which has two encryption modes of operation – a 16 bit mode and a 32 bit mode, and which is compatible only with Unix-based operating systems. Perona's technology may be applied when determining whether or not to download or install the encryption program to a target computer, as a compatibility or interoperability rule may be employed to confirm that the target computer runs a Unix platform before the encryption program is downloaded or installed. However, Perona's technology fails to inform which of the two modes of operation should be invoked upon installation of the encryption programs, as Perona teaches the use of compatibility or interoperability criteria and rules merely directed to whether or not to download or install, not whether or not to invoke functionality.

By contrast, and perhaps complementary to Perona, the claim 31 method may be used to control the mode of operation of the encryption program after a compatibility determination has been made, and the encryption program has been downloaded or installed using Perona's technology. For example, in the above example, an attribute portion within a policy file may dictate the ability of an application to utilize the 16 bit and 32 bit modes of encryption based on criteria such as, operating system version of the device running the application or security settings of a device being communicated with. Specifically, in one implementation, an attribute portion may allow the application to use 32 bit encryption when communicating with government based computers and 16 bit encryption otherwise.

Importantly, Perona simply fails to disclose rules or attributes used to determine the permissibility of an application program to use one of the functions it is otherwise capable of using. As detailed above, the Final Office Action cites Perona column 6, lines 15-23 and lines 24-36 as meeting claim limitations with respect to claim 31. The cited portion of Perona describes module requirements or constraints that must be available before the module can be loaded onto the platform,⁸ such as, requiring a signature by the National Security Agency before the module can be loaded onto the platform.⁹ The cited portions merely describe method to control downloading or installation of software (a module) but are not seen to describe or suggest indicating whether an application program may use a function, or determining whether a function represented by a retrieved attribute is permitted to be accessed by the application program.

Therefore, claim 31 defines an invention that is patentable over Drews in view of Perona, as do pending dependent claims 36-39. Accordingly, Applicants requests reconsideration and withdrawal of the imposed rejection.

Independent Claim 40 dependent claims 45-48

Similar to claim 31, independent claim 40 recites a policy file that includes attribute values, each attribute value corresponding to a policy attribute and indicating whether an application program may use a function capable of being performed by the application program and determining whether a function represented by a retrieved attribute is permitted to be accessed by the application program. For the reasons above with respect to claim 31, applicants

⁸ See Perona, Column 6, lines 20-21

⁹ See Perona, Column 6, lines 24-25

submit that the rejection of independent claim 40 and dependent claim 45-48 should be withdrawn.

Drews in View of Perona Rejection, Further in View of Anderl Rejection

Dependent claims 32-34 and 41-43 are rejected as being unpatentable over Drews in view of Perona, further in view of Anderl (WO 87/07063). Anderl does not cure the failure of Drews in view of Perona to describe or suggest the subject matter in independent claims 31 and 40, nor is Drew relied upon for such teaching or suggestion. Accordingly, Applicants requests reconsideration and withdrawal of the imposed rejection.

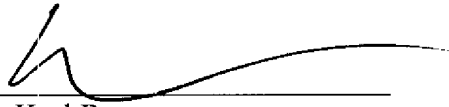
It is believed that all of the pending issues have been addressed. However, the absence of a reply to a specific rejection, issue or comment does not signify agreement with or concession of that rejection, issue or comment. In addition, because the arguments made above may not be exhaustive, there may be reasons for patentability of any or all pending claims (or other claims) that have not been expressed. Finally, nothing in this reply should be construed as an intent to concede any issue with regard to any claim, except as specifically stated in this reply, and the amendment of any claim does not necessarily signify concession of unpatentability of the claim prior to its amendment.

Applicants submit that all claims are in condition for allowance.

No fee is believed due at this time. Please apply any other charges or credits to Deposit Account No. 06-1050.

Respectfully submitted,

April 10, 2006



W. Karl Renner
Reg. No. 41,265

Customer No.: 26171
Fish & Richardson P.C.
1425 K Street, N.W.
11th Floor
Washington, DC 20005-3500
Telephone: (202) 783-5070
Facsimile: (202) 783-2331